

UNITED STATES LETTERS PATENT

APPLICANT: JAMES R. HANSEN

CERTIFICATE OF MAILING BY EXPRESS MAIL

Express Mail Label No. EL227256402US

I hereby certify under 37 CFR §1.10 that this correspondence is being deposited with the United States Postal Service as Express Mail Post Office to Addressee with sufficient postage on the date indicated below and is addressed to the Commissioner for Patents, Washington, D.C. 20231.

Date of Deposit

Signature

Typed or Printed Name of Person Signing Certificate

REPORTING THE STATE OF AN
APPARATUS TO A REMOTE COMPUTER

5

Claim To Priority

This application is a continuation-in-part application of U.S. Patent Application No. 09/627,201, filed on July 28, 2000 in the name of James R. Hansen.

Background

10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198

Summary

In general, in one aspect, the invention is directed to using a device embedded in an apparatus to report the state of the apparatus to a remote computer. This aspect
5 of the invention features detecting the state of the apparatus, generating a message that reports the state of the apparatus using a self-describing computer language, and sending the message to the remote computer. An example of a self-describing computer language is eXtensible Markup
10 Language (XML). Examples of messages that may be sent include an electronic mail (e-mail) message and a hypertext transfer protocol (HTTP) command, both containing XML code.

By virtue of the device-generated message, the remote computer can obtain the state of the apparatus even if the
15 remote computer cannot directly address the embedded device. Thus, computers that cannot communicate directly with the embedded device, such as computers that are not on the same internal network as the embedded device, can still obtain the status of the apparatus. Moreover, because the
20 state is reported using a self-describing computer language, the remote computer can interpret the state without the aid of a person. As a result, processes, such

as maintenance and the like, can be scheduled automatically for the apparatus and/or embedded device by the remote computer.

5 This aspect of the invention may include one or more of the following features. The state is indicative of an error condition in the apparatus. The error condition is a variable that deviates from an acceptable value or a predetermined range of acceptable values. The function of detecting the state includes receiving the state from the apparatus by, e.g., retrieving the state periodically from the apparatus. The function of detecting the state includes obtaining an identifier for the apparatus, the identifier relating to the state of the apparatus, and using the embedded device to read the state from the apparatus using the identifier.

10 This aspect of the invention may also include determining if the state of the apparatus has changed. The message is generated if the state of the apparatus has changed and is not generated otherwise. The function of determining if the state of the apparatus has changed includes comparing the state received from the apparatus to a previous state of the apparatus.

The message is generated using a predefined template by obtaining one or more variables relating to the apparatus and inserting the one or more variables into the template. The state of the apparatus may be included as
5 part of a body of an e-mail message or as part of an attachment to the e-mail message. The state of the apparatus may be included as part of an HTTP command.

In general, in another aspect, the invention is directed to obtaining a state of an apparatus from a
10 device, such as a controller, embedded in the apparatus. This aspect of the invention features receiving a message that reports the state of the apparatus using a self-describing computer language and extracting the state of the apparatus from the message.

15 This aspect of the invention may include one or more of the following features. The self-describing computer language is XML. The state of the apparatus is indicative of an error condition in the apparatus. The error condition is a variable that deviates from an acceptable
20 value or a predetermined range of acceptable values. The state of the apparatus is passed to a customer relationship management system. The message may be included in an HTTP

command or may be part of an e-mail.

In general, in another aspect, the invention features a system that includes first and second devices. The first device includes circuitry that generates a message

5 reporting a state of an apparatus using a self-describing computer language. The second device is in communication with the first device. The second device includes circuitry that receives the electronic mail message from the first device.

10 This aspect of the invention may include one or more of the following features. The second device receives the message from the first device and extracts the state of the apparatus from the message. The first device is embedded in the apparatus and the second device is a remote
15 computer. The message may be included in an HTTP command or may be part of an e-mail.

Other features and advantages of the invention will become apparent from the following description, including the claims and drawings.

Brief Description of the Drawings

Fig. 1 is a block diagram of a network containing a remote computer and an apparatus having an embedded device;

Fig 2 shows the format of a tag used to store state
5 variables for the apparatus;

Fig. 3 is flowchart of a process performed by the embedded device to report the state of the apparatus to the remote computer;

Fig. 4 is a flowchart of an alternative process
10 performed by the embedded device to report the state of the apparatus to the remote computer;

Fig. 5 is a flowchart of a process performed by the remote computer to interpret messages received from the embedded device; and

15 Fig. 6 is a block diagram of a network containing a remote computer and an apparatus having an embedded device the reports on the state of the apparatus using HTTP commands.

Description

Fig. 1 shows a network 10. Network 10 includes an apparatus 11 containing an embedded device 17, such as a controller (e.g., a microprocessor). Apparatus 11 is
5 connected to an internal network 12, such as a LAN. A router or modem 14 interfaces internal network 12 to an external network 15, such as the Internet, that runs TCP/IP (Transmission Control Protocol/Internet Protocol) or some other suitable protocol. Connections may be, e.g., via
10 Ethernet, wireless link, or telephone line. External network 15 contains remote computer 16, which may be a server, a personal computer (PC), or any other type of processing device. Other devices (not shown) may be
15 included on internal network 12 and external network 15.

Processing In The Embedded Device

Apparatus 11 may be any type of device or may be included in any system having functions that are monitored and controlled by embedded device 17. Among other things,
20 embedded device 17 executes software stored in memory 19 to generate and send, to remote computer 16, an e-mail message reporting the state of apparatus 11.

Software 20 includes an OPC (OLE for Process Control) server program 21, an XML (eXtensible Markup Language) processor program 24, and an e-mail program 25. E-mail program 25 is an SMTP-compliant (Simple Mail Transfer Protocol) program for sending e-mail from embedded device 17 to Internet addresses and for receiving e-mail from the Internet. E-mail program 25 operates as a mail transfer agent (MTA) for e-mail messages arriving at embedded device 17 and a mail delivery agent (MDA) for e-mail messages originating from embedded device 17. Other mail transfer protocols and programs may be also used by embedded device 17 in addition to, or instead of, those noted above.

XML processor program 24 is a program for generating XML code that reports the state of apparatus 11. XML is a self-describing computer language that defines variables and values relating to those variables. XML is self-describing in the sense that fields in the XML code identify variables and their values in the XML code. The template for XML used to generate an e-mail is as follows:

<name>temperature</name><value><##temperature##></value>,

where the "name" field identifies the name of a variable and the "value" field identifies the value of the variable that follows the "name" field. So, for the example given above, the variable is "temperature" and a value (e.g.,
5 33.8) may be inserted for that variable as follows:

<name>temperature</name><value>33.8</value>.

XML processor program 24 generates XML code having the
10 above syntax from a tag database 22 stored in memory 19.

Tag database 22 contains tags for use by XML processor
program 24 in generating XML code. Fig 2 shows an example
of a format for a tag 26, although other formats may be
used. Tag 26 contains a name field 27, a description field
15 29, a value field 30, a time stamp field 31, and an item
identifier (ID) field 32. These fields are used to obtain,
identify and store information relating to apparatus 11.

Name field 27 holds the name of a state variable for
apparatus 11, such as "temperature", and description field
20 29 provides further identification information, such as
"temperature of fluid in a tank". Value field 30 holds the
value of the state variable and time stamp field 31 holds

the time that the value in value field 30 was obtained. Value field 30 may include a variant, which is a construct that holds the value as an integer, a real number, a boolean, a character string, or some other type. Item ID
5 field 32 holds an identifier that corresponds to hardware that is being monitored within apparatus 11. The identifier corresponds to a register location or to some other storage area of apparatus 11 that contains the value for field 30. For example, if embedded device 17 is in a
10 robotics system, item ID field 32 might correspond to a register in the robotics system that contains a velocity or position of a robotic arm.

OPC server program 21 reads item IDs from field 32 and uses those item IDs to read variable values from
15 corresponding hardware storage areas 34. OPC server program 21 implements an industrial automation protocol, such as MODBUS TCP, to communicate with the apparatus hardware. The system is not limited to use with the MODBUS protocol or with OPC server program 21; any drivers or
20 computer programs may be used to read the state variable values from the hardware. Once a state variable value has been read, OPC server program 21 inserts the variable value

into field 30 of the appropriate tag.

Fig. 3 shows a process 36 for reporting the state of apparatus 11 to remote computer 16 using e-mail. In this embodiment, process 36 is implemented by OPC server program 21, XML processor program 24, e-mail program 25, and system software (not shown) executing in embedded device 17. The system software may include an operating system or other programs that control the background operation of embedded device 17.

Process 36 detects (301) the state of apparatus 11. The state may be indicative of an error condition (described below) within apparatus 11 or it may simply be state variables of apparatus 11 that are obtained at a particular time. To detect the state of apparatus 11, OPC server program 21 polls the hardware in apparatus 11 periodically. To perform this polling, OPC server program 21 obtains (301a) an item ID from tag database 22 and reads (301b) the value of a state variable that corresponds to the item ID from the appropriate hardware storage location. Process 36 may report the value to the remote computer as is or, alternatively, process 36 may use the value to identify and report an error condition in the hardware. A

process for reporting error conditions is described below.

Process 36 generates (302) an e-mail message reporting the value of state variable(s) for apparatus 11.

Specifically, XML processor program 24 retrieves both the
5 name of each state variable and the value of the state
variable from the appropriate tag(s) in tag database 22.
Other variables may also be retrieved from tag database 22
including the time stamp, description, and whatever other
variables are stored in tag database 22. Which information
10 is retrieved is pre-set in XML processor program 24. The
retrieved variables are used by XML processor program 24 to
generate XML code for an e-mail to remote computer 16.

XML processor program 24 may generate the XML code "on
the fly", meaning without the use of a template. In this
15 case, a blank XML file is populated with the retrieved
variables in XML format by XML processor program 24.
Alternatively, XML processor program 24 may generate the
XML code using a pre-defined and formatted template. The
template may be obtained by XML processor program 24, e.g.,
20 from memory 19 or a remote storage location (not shown).
For example, the template may contain formatting similar to
that shown above, namely:

<name>temperature</name><value><##temperature##></value>.

To generate the XML code from the template, XML processor
5 program 24 scans through the template and inserts state
variable value(s) retrieved from tag database 22, where
appropriate. XML processor program 24 may generate the XML
code periodically, depending upon how often e-mails are to
be sent to the remote computer. Alternatively, tag manager
10 software (not shown) may be included to provide newly-
received tag variables to XML processor program 24. In
this case, XML processor program 24 generates the XML code
when it receives the new tag variables.

The resulting XML code may be part of the body of an
15 e-mail or it may part of an attachment to an e-mail. The
e-mail also contains a unique identifier, such as a code
(e.g., serial number or identifier), that identifies
embedded device 17 to remote computer 16. E-mail program
25 obtains the XML code from XML processor program 24 and
20 sends it to remote computer 16 as part of the e-mail
message. E-mail program 25 obtains the code periodically,
depending upon the frequency at which e-mails are to be

sent to the remote computer. The frequency is set
beforehand in embedded device 17. The address of the
remote computer may be registered with e-mail program 25
beforehand. Typically, the address/remote computer will be
5 that of an entity that requires information about apparatus
11. For example, the entity may be a manufacturer of the
apparatus, a plant monitoring system, or the like. The e-
mail program sends the message to router/modem 14, which
transfers it via external network 15 to remote computer 16.
10 Then, the e-mail message is processed as described below.

The foregoing describes the case where embedded device
17 simply reports the state of apparatus 11 to remote
computer 16 periodically. Alternatively, embedded device
17 may report the state to remote computer 16 only when an
15 error condition or "alarm" is detected.

Fig. 4 shows a process 40 by which embedded device 17
detects error conditions in apparatus 11 and sends an e-
mail message to remote computer 16 when an error condition
is detected. Process 40 detects (401) the state of
20 apparatus 11, where, as above, "state" refers to tag
variable values for apparatus 11. Detection (401) is
performed in the same manner as process 36; therefore, a

description is omitted here. Once process 36 has obtained the state of apparatus 11, process 36 determines (402) if that state represents an error condition.

To detect an error condition, process 40 may compare
5 an obtained state variable value to a predetermined acceptable value or a range of predetermined acceptable values. If the state variable value is outside the range of, or deviates considerably from, the acceptable value(s), then process 40 knows that an error condition is present.
10 Alternatively, process 40 may store each state variable value in memory 19 as it is obtained, and compare each newly-received state variable value to one or more stored state variable values. If the new state variable value deviates by more than a predetermined amount from the
15 stored value(s), process 40 knows that an error condition is present/has occurred.

An error condition may be based on a single state variable value or it may be based on some combination of two or more state variable values. For example, if
20 embedded device 17 is in manufacturing equipment that monitors both a level of fluid in a tank and a temperature of that fluid, an error condition may only be present if

both the fluid level and the temperature exceed preset values. In this example, therefore, if only one state variable exceeds its corresponding preset value, then no error condition is present/has occurred.

5 If process 40 detects (402) an error condition, process 40 generates (403) an e-mail message and sends (404) the e-mail message to remote computer 16. The functions of generating and sending an e-mail message are performed as described above with respect to process 36; therefore, detailed descriptions are omitted here. When
10 generating the e-mail message, e-mail program 25 may place the state variable(s) that caused the error condition in the "subject" line of the e-mail. If process 40 does not detect (402) an error condition, an e-mail message is not
15 sent, whereafter process 40 returns to 401.

 XML processor program 24 may maintain a log of error conditions in memory 19. This error condition "history" may be provided along with each new e-mail message. The history may relate to a particular state variable or to
20 more than one state variable. For example, if the error condition pertains to temperature, XML processor program 24 may include the error condition history for temperature in

the e-mail. If the error condition pertains to both temperature and tank level, XML processor program 24 may include the error condition history for both temperature and tank level in the e-mail. If a template is used to
5 generate the e-mail message, portion(s) of that template may be reserved for error condition history.

Processes 36 and 40 can be combined to generate an e-mail periodically that reports the state of apparatus 11 to remote computer 16 even if no error conditions have been
10 detected in apparatus 11, and that also flags any error conditions if any have been detected. XML processor program 24 adds an indicator or the like next to state variable values that correspond to error conditions.

Processes 36 and 40 may be executed by embedded device
15 17 to monitor and report on any type of state variables in any type of apparatus. For example, processes 36 and 40 may detect state variable values relating to conveyor belt speed, current and/or voltage in electronic devices, tank fluid levels, input/output sensors, and the like.

20 Processes 36 and 40 may detect state variable values through a programmable logic controller (PLC) that is connected to one or more other devices. A PLC includes

plug-in cards for each device that obtain and store device state variable values. OPC server program 21 communicates with these plug-in cards to obtain the device state variable values for generating e-mails as described above.

5 E-mails generated by processes 36 and 40 report the state of apparatus 11 using a self-describing computer language, such as XML; however, other types of self-describing computer languages may be used. In addition, other text and/or images may be included in the e-mails, if
10 desired and appropriate under the circumstances. Described below is a process that is performed by remote computer 16 to interpret e-mails received from embedded device 17.

Processing In The Remote Computer

15 Remote computer 16 contains a controller 41 for executing software stored in memory 42. Among this software is e-mail program 44, XML parser 45, and customer relationship management (CRM) system software 46.

As in embedded device 17, e-mail program 44 is an
20 SMTP-compliant program for receiving e-mail from embedded device 17 and other such devices. E-mail program 44 operates as a mail transfer agent (MTA) for e-mail messages

arriving at remote computer 16 and a mail delivery agent (MDA) for e-mail messages originating from remote computer 16. E-mail program 44 uses the same protocol as e-mail program 25 in embedded device 17.

5 XML parser 45 parses XML code in a received e-mail to extract variable values, including an identifier for apparatus 11. XML parser 45 recognizes field names, such as "name" and "value" from above and extracts corresponding state variable values from those fields. That is, XML
10 parser 45 knows the syntax of XML. Knowing this, XML parser 45 is able to extract variable names from the "name" fields, corresponding variable values from the "value" fields, and any other information in the XML code.

15 XML parser 45 passes the state variable values, along with appropriate identifiers, to customer relationship management system software 46 or whatever other software or database requires/uses those state variable values.

Fig. 5 shows how an e-mail from embedded device 17 is processed (43). Once an e-mail has been received (501)
20 from embedded device 17, XML parser 45 extracts (502) the state variable values of apparatus 11 from the e-mail. For example, XML parser 45 may extract tank levels, temperature

values, etc., of apparatus 11 monitored by embedded device 17. The state variable values may be indicative of error conditions in apparatus 11, as defined above, or simply state variables for apparatus 11 obtained at a given point
5 in time.

XML parser 45 passes (503) the state variable values, i.e., the state of apparatus 11, to customer relationship management system software 46. Customer relationship management system software 46 uses these state variable
10 values, e.g., to schedule maintenance for apparatus 11 if necessary, to provide software upgrades to apparatus 11, or for any other purpose. Because the XML code in the e-mail is readable by XML parser 45, reporting and scheduling by customer relationship management system software 46 can be
15 done automatically. It is noted that e-mail program 44 may still forward an e-mail to a customer representative, technician, or the like, particularly if an e-mail contains human-readable text.

The software on remote computer 16 is not limited to
20 that shown in Fig. 1. For example, XML parser 45 may be replaced by a parser that is capable of parsing/reading other types of computer code, depending upon the code that

is used in the received e-mail. Likewise, the parsed variables can be passed to software other than customer relationship management system software 46. For example, the variables can be stored in a database 47 for later use.

5

Alternative Embodiment

Referring to Fig. 6, a network 60 is shown on which an alternative embodiment of the invention is implemented.

Network 60 is identical to network 10, except that e-mail program 25 in apparatus 11 is replaced by Web client 61 and e-mail program 44 in remote computer 16 is replaced by Web server 62. This alternative configuration allows embedded device 17 to transfer messages to remote computer 16 as HTTP commands rather than e-mails.

The HTTP command may be an HTTP POST command, although other HTTP commands, such as an HTTP GET command, may instead be used. An example of an HTTP POST command that uses XML code to report the status of a fictitious "widget" apparatus is as follows:

20

POST /CONTROL HTTP/1.1
Host: www.acme.com
Content-Type: text/xml
Content-length: nnn

```
5  <?xml version="1.0"?>
   <root xmlns="urn:schemas-upnp-org:device-1-0">
     <specVersion>
       <major>1</major>
10    <minor>0</minor>
     </specVersion>
     <device>
       <deviceType>urn:www-acme-
15 com:device:Widget:3</deviceType>
       <friendlyName>Widget</friendlyName>
       <manufacturer>Acme Industries</manufacturer>
       <modelName>Widget</modelName>
       <modelName>3</modelName>
       <serialNumber>53266D</serialNumber>
20    <UDN>uuid:4A89EA70-73B4-11d4-80DF-0050DAB7BAC5</UDN>
     </device>
   </root>
   <parameters>
     <Airflow xsd:type="integer">378</Airflow>
25    <Humidity xsd:type="double">46.7</Humidity>
     <Motor xsd:type="integer">1500</Motor>
     <Vent xsd:type="integer">4</Vent>
   </parameters>
   <alarms>
30    <Temperature>
       <description>Room temperature is above
83F</description>
       <severity>300</severity>
       <status>high</status>
35    </Temperature>
   </alarms>
```

XML is a self-describing computer language in the sense that fields in the XML code identify variables and their values in the XML code. For example, as shown in the

5

10

15

25

5 <parameters>
 <Airflow xsd:type="integer">378</Airflow>
 <Humidity xsd:type="double">46.7</Humidity>
 <Motor xsd:type="integer">1500</Motor>
 <Vent xsd:type="integer">4</Vent>
 </parameters>
 <alarms>
10 <Temperature>
 <description>Room temperature is above
 83F</description>
 <severity>300</severity>
 <status>high</status>
 </Temperature>
15 </alarms>

Thus, the state of the widget includes information on its
airflow, humidity, motor and vent settings, temperature,
severity of the temperature, and temperature status.

20 Different information from that shown may be included in
the HTTP POST command.

Referring back to Figs. 3, 4 and 5, in this embodiment
the operation of processes 36, 40 and 43 is identical to
that described above, except that, in all steps, the e-mail
25 message is replaced by an HTTP command. In apparatus 11,
the HTTP command is generated by Web client 61 based on
data provided by XML processor 24. This XML data is the
same as that used above with e-mail program 25. Embedded
device 17 sends the HTTP command to remote computer 16,
30 where it is received by Web server 62 and then processed by

XML parser 45. Thereafter, processing proceeds as above.

Architecture

Processes 36, 40 and 43 are not limited to use with
5 the hardware/software configuration of Fig. 1; they may
find applicability in any computing or processing
environment. Processes 36, 40 and 43 may be implemented in
hardware (e.g., an ASIC {Application-Specific Integrated
Circuit} and/or an FPGA {Field Programmable Gate Array}),
10 software, or a combination of hardware and software.

Processes 36, 40 and 43 may be implemented using one
or more computer programs executing on programmable
computers that each includes a processor, a storage medium
readable by the processor (including volatile and non-
15 volatile memory and/or storage elements), at least one
input device, and one or more output devices.

Each such program may be implemented in a high level
procedural or object-oriented programming language to
communicate with a computer system. Also, the programs can
20 be implemented in assembly or machine language. The
language may be a compiled or an interpreted language.

Each computer program may be stored on a storage medium or device (e.g., CD-ROM, hard disk, or magnetic diskette) that is readable by a general or special purpose programmable computer for configuring and operating the
5 computer when the storage medium or device is read by the computer to perform processes 36, 40 and 43.

Processes 36, 40 and 43 may also be implemented as a computer-readable storage medium, configured with a computer program, where, upon execution, instructions in
10 the computer program cause the computer to operate in accordance with processes 36, 40 and 43.

Other embodiments not described herein are also within the scope of the following claims. For example, e-mail or http messages sent from apparatus 11 to remote computer 16
15 may be queued (e.g., stored in memory 19) and then retrieved and sent out at a later time. Queuing messages reduces message loss resulting from intermittent system failures.

What is claimed is: